

# Recurrent Support Vector Regression for a Nonlinear ARMA Model with Applications to Forecasting Financial Returns

Shiyi Chen\*

Kiho Jeong\*\*

Wolfgang K. Härdle\*\*\*



\* Fudan University, P.R.China

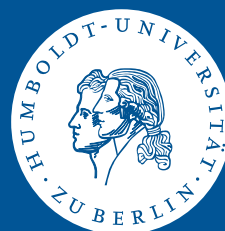
\*\* Kyungpook National University, Korea

\*\*\* Humboldt-Universität zu Berlin, Germany

This research was supported by the Deutsche  
Forschungsgemeinschaft through the SFB 649 "Economic Risk".

<http://sfb649.wiwi.hu-berlin.de>  
ISSN 1860-5664

SFB 649, Humboldt-Universität zu Berlin  
Spandauer Straße 1, D-10178 Berlin



# Recurrent Support Vector Regression for a Nonlinear ARMA Model with Applications to Forecasting Financial Returns \*

Shiyi Chen,<sup>†</sup> Kiho Jeong,<sup>‡</sup> Wolfgang K. Härdle<sup>§</sup>

## Abstract

Motivated by the recurrent Neural Networks, this paper proposes a recurrent Support Vector Regression (SVR) procedure to forecast nonlinear ARMA model based simulated data and real data of financial returns. The forecasting ability of the recurrent SVR is compared with three competing methods, MLE, recurrent MLP and feedforward SVR. Theoretically, MLE and MLP only focus on fit in-sample, but SVR considers both fit and forecast out-of-sample which endows SVR with an excellent forecasting ability. This is confirmed by the evidence from the simulated and real data based on two forecasting accuracy evaluation metrics (NSME and sign). That is, for one-step-ahead forecasting, the recurrent SVR is consistently better than the MLE and the recurrent MLP in forecasting both the magnitude and turning points, and really improves the forecasting performance as opposed to the usual feedforward SVR.

*Keywords:* Recurrent Support Vector Regression; MLE; recurrent MLP; nonlinear ARMA; financial forecasting

*JEL classification:* C45, F37, F47

---

\*This work was supported by Deutsche Forschungsgemeinschaft through the SFB 649 “Economic Risk”. Shiyi Chen was also sponsored by Kyungpook National University Graduate Scholarship for Excellent International Students and Shanghai Pujiang Program.

<sup>†</sup>(Corresponding author) China Center for Economic Studies (CCES), Fudan University, 220 Handan Road, 200433 Shanghai, P.R.China; e-mail: shiyichen@fudan.edu.cn; TEL:+86 21-6564-2050

<sup>‡</sup>School of Economics and Trade, Kyungpook National University, Daegu 702-701, Korea; Email: khjeong@knu.ac.kr; TEL: +82 53-950-5416

<sup>§</sup>Center for Applied Statistics and Economics (CASE), Humboldt-Universität zu Berlin, Spandauer D-10178, Berlin, Germany; Email: haerdle@wiwi.hu-berlin.de; TEL: +49 30-20-93-56-31

# 1 Introduction

This paper considers financial returns forecasting in the framework of a univariate Autoregressive Moving Average (ARMA) model by using the proposed recurrent  $\varepsilon$ -SVR approach. For more than two decades, the linear ARMA model estimated by Maximum Likelihood Estimation (MLE) has been a popular approach for forecasting nonstationary time series. This implies that the analyzed variables should satisfy the normal assumption and have large sample. However, it has been widely accepted that the returns of a variety of financial variables are not linearly predictable in general and the phenomenon of volatility clustering in it leads to the violation of the normal assumption, as a result of which, the linear ARMA model by MLE usually tends to provide poor forecasting performance (Priestley (1988), Box, Jenkins & Reinsel (1994), Niemira & Klein (1994) and Hamilton (1997)). Thus, some nonlinear, nonparametric alternative approaches are proposed and adopted to estimate the time series models, the prevailing representative among them is the Artificial Neural Network (ANN). Plentiful of studies on ANN denote that ANN approach outperforms traditional MLE in forecasting financial time series and, particularly, the recurrent ANN with richer dynamic structure could capture more characteristics of data in the generalization period than the feedforward one ((Kuan, 1995), (Wu, 1995), (Tian, Juhola & Grönfors, 1997), (Lisi & Schiavo, 1999), (Ashok & Mitra, 2002), (Gaudart, Giusiano & Huiart, 2004), (Kamruzzaman & Sarker, 2004)), but some indicate mixed or opposite results ((Adya & Collopy, 1998); (Kanas, 2003)). While the ANN is theoretically better in estimating nonlinear finite samples without invoking a probabilistic distribution, however, it has been criticized to be vulnerable to the over-fitting problem which usually leads to a local optimum and to the empirical risk minimization, same as the MLE<sup>1</sup>, the latter of which results in good fit and poor forecast out-of-sample. To avoid the theoretical pitfalls of the MLE and ANN in forecasting area, fortunately, Vapnik (1995) ,Vapnik (1997) has successfully developed a novel nonparametric function approximator, the Support Vector Machine (SVM), which is computationally powerful in the sense that it allows for (1) finite and infinite sample; (2) no prior distribution assumption; and (3) minimizing the structural risk as opposed to empirical risk employed by MLE and ANN, which endows SVM with an excellent generalization, or forecasting, ability out-of-sample and is the biggest advantage of SVM among all alternatives (We refer to next section for a detailed explanation).

---

<sup>1</sup>For MLE, maximizing the joint probability density function amounts to minimizing the sum of residual squares, i.e., minimizing the empirical risk, which is precisely the OLS approach.

SVM was originally developed for classification problems (SVC) and then extended to regression problems (SVR). Recently, SVM has been successfully applied to financial variable classification and financial time series forecasting, for example, see Trafalis & Ince (2000), Cao & Tay (2001), Gestel, Suykens, Baestaens, Lambrechts, Lanckriet, Vandaele, Moor & Vandewalle (2001), Yang, Chan & King (2002), Härdle, Moro & Schäfer (2005), Härdle, Moro & Schäfer (2006), Espinoza, Suykens & Moor (2006) and Lee, Chiu, Chou & Lu (2006), to name a few. As Haykin (1999) argued earlier, the present studies on SVM mostly focus on the feedforward direction and the previous application literatures of the SVR based time series forecasting only consider the dynamic systems of nonlinear Autoregressive (AR) model. In a context of networks, these systems do not have feedback loops from the output layer to the input layers. It is well known that recurrent ANN, networks with feedbacks, can characterize the behavior of time series variables with richer dynamic structures and have more potential of reducing the memory requirement significantly than the feedforward one ((Kuan, Hornik & White, 1994); (Kuan, 1995); (Kuan & Liu, 1995)). Suykens & Vandewalle (2000) and Suykens, Gestel, Brabanter, Moor & Vandewalle (2002) extend the recurrent networks to support vector machine and proposed a new recurrent least squares SVM (LS-SVM) procedure. Their studies reveal that recurrent LS-SVM can forecast time series well, even on relatively small training data sets.

Motivated by the recurrent ANN and LS-SVM, in this paper, we propose a new  $\varepsilon$ -insensitive loss based Support Vector Regression (SVR) procedure with the addition of a global feedback connection from the output layer to the input space. In terms of the terminology of the recurrent LS-SVM, we refer to the proposed procedure as a recurrent  $\varepsilon$ -SVR and to the standard SVR as a feedforward SVR. The difference between the recurrent LS-SVM and our recurrent  $\varepsilon$ -SVR is that they use different empirical loss function; the former adopts the mean square error (MSE), the later uses the  $\varepsilon$ -insensitive error which can lead to sparseness solutions (see Section 2 for details). As empirical application, the proposed recurrent  $\varepsilon$ -SVR procedure is applied to forecasting the ARMA model for the simulated data (linear ARMA series and nonlinear Lorenz series) and the real data of financial returns (Canadian Dollar against the U.S. dollar (CAD) exchange rates and New York Stock Exchange<sup>TM</sup> (NYSE) composite stock index). To examine the sensitivity of the recurrent  $\varepsilon$ -SVR with respect to free parameters, we experiment with three free parameters,  $\varepsilon$ ,  $C$  and  $\sigma^2$  by using cross validation method. The iterative epochs of recurrent  $\varepsilon$ -SVR procedure are also described in Section 2 and illustrated by simulation. The forecasting performance among the recurrent SVR, MLE, recurrent ANN and the feedforward SVR is compared

by using two forecasting evaluation metrics (NMSE and sign) in one-step-ahead forecasting horizon.

This paper is organized as follows. Section 2 introduces the theory of standard SVR and proposes the recurrent  $\varepsilon$ -SVR procedure. Section 3 specifies the empirical modeling and forecasting scheme. Section 4 compares the forecasting performance of all candidates by using the simulated and real data, in which the parameters selection and iterative process are illustrated in detail. The conclusion is presented in Section 5.

## 2 Support Vector Regression (SVR)

### 2.1 Principle of standard $\varepsilon$ -SVR

The Support Vector Machines for Regression (SVR) originates from Vapnik's statistical learning theory (Vapnik (1995) ,Vapnik (1997)), which has the design of a feedforward network with an input layer, a single hidden layer of nonlinear units and an output layer and formulates the regression problem as a quadratic programming (QP) problem (Haykin, 1999). SVR estimates a function by nonlinearly mapping the input space into a high dimensional hidden space and then running the linear regression in the output space (see Figure 1). Thus, the linear regression in the output space corresponds to a nonlinear regression in the low dimensional input space. And the theory denotes that if the dimensions of feature space (or hidden space) are high enough, SVR may approximate any nonlinear mapping relations. As the name implies, the design of the SVR hinges upon the extraction of a subset of the training data that serves as support vectors which represent a stable characteristic of the data.

Given a training data set  $\{(x_t, y_t)\}_{t=1}^T$ , where inputs vector  $x_t \in \mathbb{R}^p$  and output scalar  $y_t \in \mathbb{R}^l$ . In classification problem, the variable  $y$  only takes two values,  $-1$  and  $1$ ; while in regression  $y$  can take any real values. Indeed, the desired response  $y$ , known as a 'teacher', represents the optimum action to be performed by the SVR. We aim at finding a sample regression function  $f(x)$ (or denoted by  $\hat{y}$ ) as below to approximate the latent, unknown decision function  $g(x)$ .

$$f(x) = w^\top \phi(x) + b, \quad (1)$$

where

$$\begin{aligned} \phi(x) &= [\phi_1(x), \dots, \phi_l(x)]^\top \\ w &= [\omega_1, \dots, \omega_l]^\top. \end{aligned}$$

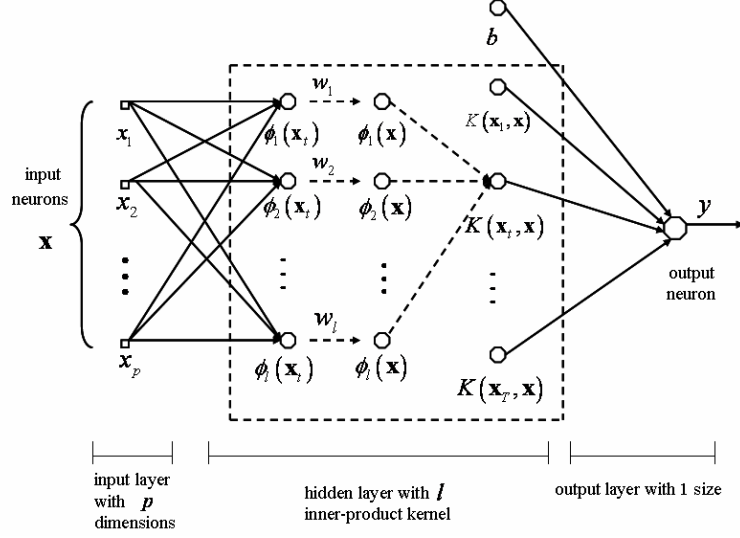


Figure 1: Architecture of Support Vector Machines.

The  $\phi(x)$  is known as the nonlinear transfer function which represents the features of the input space and projects the inputs into the feature space. The dimension of the feature space is  $l$  which is directly related to the capacity of the SVR to approximate a smooth input-output mapping; the higher the dimension of the feature space, the more accurate the approximation will be. Parameter  $w$  denotes a set of linear weights connecting the feature space to the output space, and  $b$  is the threshold.

To get the function  $f(x)$ , the optimal  $w^*$  and  $b^*$  have to be estimated from the data. Firstly, we define a linear  $\varepsilon$ -insensitive loss function  $L_\varepsilon$ , originally proposed by Vapnik (1995).

$$L_\varepsilon(x, y, f(x)) = \begin{cases} |y - f(x)| - \varepsilon & \text{for } |y - f(x)| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

This function indicates the fact that it does not penalize errors below  $\varepsilon$ . The training points within the  $\varepsilon$ -tube have no loss and do not provide any information for decision. Therefore, these points do not appear in the decision function  $f(x)$ . Only those data points located on or outside the  $\varepsilon$ -tube will serve as the support vectors to finally used to construct the  $f(x)$ . The sparseness property of algorithm results only from the  $\varepsilon$ -insensitive loss function and greatly simplifies the computation of SVR. Thus, SVR based on it is also called  $\varepsilon$ -SVR, which is different from the other loss functions such

as (mean) squared errors (MSE). The nonnegative slack variables,  $\xi$  and  $\xi'$  (below or above the  $\varepsilon$ -tube, or denoted together by  $\xi^{(l)}$ ; see Figure 2) are employed to describe this kind of  $\varepsilon$ -insensitive loss, that is, the loss of error on training points out of the  $\varepsilon$ -tube.

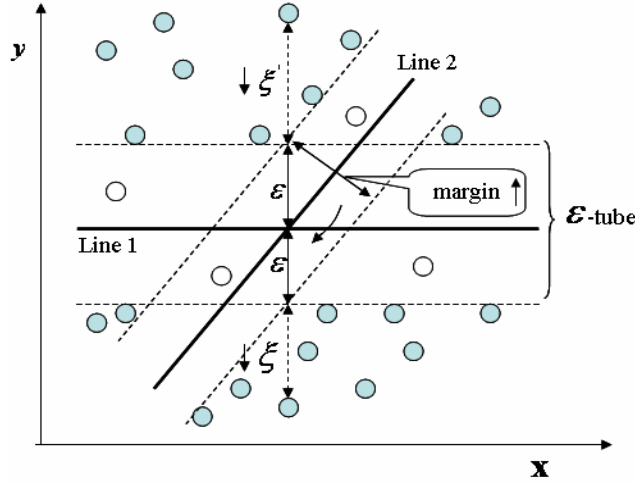


Figure 2: Principle of Structural Risk Minimization of  $\varepsilon$ -SVR.

The derivation of SVR follows the principle of structural risk minimization that is rooted in VC dimension theory. Structural risk is the upper boundary of empirical loss, denoted by  $\varepsilon$ -insensitive loss function, plus the confidence interval (or called margin), which is constructed in equation 3. The primal constrained optimization problem of  $\varepsilon$ -SVR is obtained below:

$$\min_{w \in \mathbb{R}^l, \xi^{(l)} \in \mathbb{R}^{2T}, b \in \mathbb{R}} C(w, b, \xi_t, \xi'_t) = \frac{1}{2} \|w\|^2 + C \sum_{t=1}^T (\xi_t + \xi'_t) \quad (3)$$

$$s.t. \quad w^\top \phi(x_t) + b - y_t \leq \varepsilon + \xi_t, \quad t = 1, 2, \dots, T \quad (4)$$

$$y_t - w^\top \phi(x_t) - b \leq \varepsilon + \xi'_t, \quad t = 1, 2, \dots, T \quad (5)$$

$$\xi_t \geq 0, \xi'_t \geq 0, \quad t = 1, 2, \dots, T. \quad (6)$$

The formulation of the cost function  $C(w, b, \xi_t, \xi'_t)$  in equation (3) is in perfect accord with the principle of structural risk minimization, which is illustrated in Figure 2 (in which the dark circles are data points extracted as

support vectors). In equation 3, the first term indicates the Euclidean norm of the weight vector  $w$  ( $\|w\|^2 = w^\top w$ ) and measures the function flatness; the minimization of it is related to the maximization of the margin of separation ( $2/\|w\|$ ), i.e., maximizing the generalization ability. The second term represents the empirical risk loss determined by the  $\varepsilon$ -insensitive loss function and is similar to the sum of residual squares in the objective function of MLE and ANN. Finally, SVR obtains the tradeoff between the two terms; as a result, it not only fits the historical data well but forecasts the future data excellent. As shown in Figure 2, both regression lines 1 and 2 can classify the data points correctly and then minimize the empirical loss; however, the margin of generalization of the two lines are different in which the regression line 1 has the largest margin. It is the special design of minimizing the structural risk that endows SVR with the excellent forecasting ability among all candidates. Evgeniou, Poggio, Pontil & Verri (2002) also denoted that minimization of an empirical error only is both ill-posed and not necessarily leading to models with good predictive capabilities, thus, one needs to minimize a structural risk. In addition, the convex quadratic programming and linear restrictions in above primal problem ensure that SVR can always obtain the global unique optimal solution, which is different from the usual networks that easily get trapped in local minima. The penalty parameter  $C > 0$  controls the penalizing extent on the sample which lie out of  $\varepsilon$ -tube. Both  $\varepsilon$  and  $C$  must be selected by the user.

The corresponding dual problem of the  $\varepsilon$ -SVR can be derived from the primal problem by using the Karush-Kuhn-Tucker conditions as follows.

$$\min_{\alpha_t^{(l)} \in \mathbb{R}^{2T}} \frac{1}{2} \sum_{s=1}^T \sum_{t=1}^T (\alpha'_s - \alpha_s)(\alpha'_t - \alpha_t) K(x_s \cdot x_t) + \varepsilon \sum_{t=1}^T (\alpha'_t + \alpha_t) - \sum_{t=1}^T (\alpha'_t - \alpha_t) \quad (7)$$

$$s.t. \quad \sum_{t=1}^T (\alpha_t - \alpha'_t) = 0 \quad (8)$$

$$0 \leq \alpha_t, \alpha'_t \leq C, \quad s, t = 1, 2, \dots, T. \quad (9)$$

where,  $\alpha_t$  and  $\alpha'_t$  (or  $\alpha_t^{(l)}$ ) are the Lagrange multipliers. The dual problem can be solved more easily than the primal problem ((Scholkopf & Smola, 2001), (Deng & Tian, 2004)). Making use of any solution,  $\alpha_t$  and  $\alpha'_t$ , the optimal solutions of primal problem can be calculated, in which,  $w^*$  is unique and expressed as follows:

$$w^* = \sum_{t=1}^T (\alpha'_t - \alpha_t) \phi(x_t) \quad (10)$$



However,  $b^*$  is not unique and formulated in terms of different cases. If  $i \in \{t | \alpha_t \in (0, C)\}$ ,

$$b^* = y_i - \sum_{t=1}^T (\alpha'_t - \alpha_t) K(x_t \cdot x_i) + \varepsilon \quad (11)$$

if  $j \in \{t | \alpha'_t \in (0, C)\}$ ,

$$b^* = y_j - \sum_{t=1}^T (\alpha'_t - \alpha_t) K(x_t \cdot x_j) - \varepsilon \quad (12)$$

The cases of both  $i, j \in \{t | \alpha_t^{(i)} = 0\}$  and  $i, j \in \{t | \alpha_t^{(i)} = C\}$  rarely occur in reality.

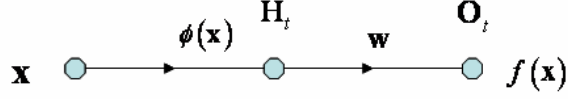
Thus, the regression decision function  $f(x)$  will be computed by using of  $w^*$  and  $b^*$  in the following forms:

$$\begin{aligned} f(x) &= w^{*T} \phi(x) + b^* \\ &= \sum_{t=1}^T (\alpha'_t - \alpha_t) \phi^T(x_t) \phi(x) + b' \\ &= \sum_{t=1}^T (\alpha'_t - \alpha_t) K(x_t, x) + b^* \end{aligned} \quad (13)$$

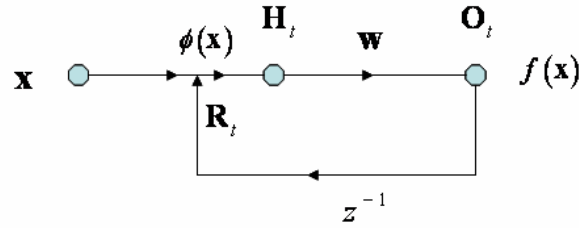
where  $K(x_t, x) = \phi^T(x_t) \phi(x)$  is the inner-product kernel function. In fact, the SVR theory considers only the form of  $K(x_t, x)$  in the feature space without specifying  $\phi(x)$  explicitly and without computing all corresponding inner products. Therefore, the kernel function greatly reduces the computational complexity of high dimensional hidden space and becomes the crucial part of SVR. The function which satisfies Mercer theorem can be chosen as the SVR kernel. In this paper the chosen kernel is widely used Gaussian kernel, or called radial based function (RBF) kernel which offers a way to measure proximity between two data points and is expressed as below.

$$K(x_t, x) = \exp\left(\frac{-\|x - x_t\|^2}{2\sigma^2}\right) \quad (14)$$

where  $\sigma^2$  is the kernel width which implicitly controls the complexity of the feature space and the solution (the higher the  $\sigma^2$  is, the lower is the complexity). For the Gaussian kernel, the explicit expression of nonlinear transformation function  $\phi(x)$  is unknown, and the corresponding feature dimension  $l$  is infinite.



(a) Signal-flow graph of a feedforward SVR



(b) Signal-flow graph of a single-loop recurrent SVR

Figure 3: Signal-flow graphs of feedforward and recurrent SVR.

## 2.2 Algorithm of Recurrent $\varepsilon$ -SVR

As Haykin (1999) said, the SVR described in Subsection 2.1 usually appears in the design of a simple feedforward network in which an input layer of source nodes projects onto an output layer of computation node, but not vice versa, see Figure 3(a). This process is known as feedforward SVR. If the in-sample fitting errors are white noise, or do not display autocorrelation, the feedforward SVR is efficient in the sense that they can be utilized to estimate AR(p) model directly. Let  $O_t$  and  $H_t$  represent the single-output and  $l$  hidden unit activations. Symbolically, we have

$$O_t = \psi(w^\top H_t + b); H_t = \phi(x_t) \quad (15)$$

where  $x_t = \{x_{t,i}\}_{i=1}^p = \{y_{t-i}\}_{i=1}^p$ . Note that  $\psi$  and  $\phi$  are vector-valued functions and represent the identity function and the transfer function to produce Gaussian kernel, respectively.

If it is not the case, the information reflected behind the errors should be utilized to improve the estimating power of the model, thus, ARMA model,

i.e., introducing the error terms (MA part) into the AR model, becomes reasonable. To estimate the ARMA model, a feedback process of  $\varepsilon$ -SVR with unobservable MA part as inputs, not addressed before our application,<sup>2</sup> has to be described - which distinguishes itself from feedforward SVR in that it has at least one feedback loop (see Figure 3(b)). In this paper, we abuse terminology and refer to this process as “recurrent  $\varepsilon$ -SVR”. The feedback loops involve the use of particular branches composed of *one-delay operator*,  $z^{-1}$ , which result in nonlinear dynamical behavior and have a profound impact on the learning capability of SVR. Thus, the recurrent  $\varepsilon$ -SVR will capture more dynamic characteristics of  $y_t$  than does feedforward SVR.

Let  $R_t$  denotes one-delayed internal feedbacks. Then, the recurrent  $\varepsilon$ -SVR can be represented in the following generic form

$$O_t = \psi(w^\top H_t + b); H_t = \phi(x_t, R_t) \quad (16)$$

where  $x_t = \{y_{t-i}\}_{i=1}^p$ .  $R_t$  is chosen to be  $O_{t-1}$ ; that is, the recurrent process has output feedbacks rather than hidden unit activations feedbacks. Thus,  $R_t$  can be expressed as

$$R_t = \tau(x_{t-1}, R_{t-1}; w, b) \quad (17)$$

with  $\tau$  also a vector-valued function.

If  $R_t = 0$ , the process simply reduces to a feedforward SVR, in which the finite lagged responses are used as inputs to capture dynamics. This approach suffers the drawback that the correct lag length needed is typically unknown and somewhat difficult to determine. On the one hand, the finite lagged dependent variables may not be enough to capture certain temporal structures, especially, that depend on a long history of targets. On the other hand, storing all the past information in memory is practically implausible. The case is similar to building a linear AR model with finite  $p$  lags. This deficiency could be circumvented by our device of recurrent SVR. The feedback variable  $R_t$  will serve as a memory device to store past information compactly. That is,

$$R_t = \tau(x_{t-1}, \tau(x_{t-2}, R_{t-2}; w, b); w, b) = \dots = v(x_{t-1}, x_{t-2}, \dots, x_1; w, b) \quad (18)$$

Thus, the output of recurrent SVR can be write in the following feedforward form

$$O_t = \psi(w^\top \phi(x_t, R_t) + b) \quad (19)$$

---

<sup>2</sup>Suykens & Vandewalle (2000) proposed the algorithm of recurrent least squares SVM. The difference between the two recurrent SVM algorithm is their sparseness solutions.

$$\begin{aligned}
&= \kappa(x_t, R_t(w, b); w, b) \\
&= f(x_t, x_{t-1}, x_{t-2}, \dots, x_1; w, b)
\end{aligned}$$

As  $R_t$  depends on the entire history of inputs, introducing recurrent variable  $R_t$  with the contraction mapping requirement of  $\tau$  to a feedforward SVR is similar to adding invertible moving average term to an AR model. Therefore, a recurrent SVR may be interpreted as a parsimonious model which incorporates all the past inputs without storing all of them in memory. That is, in our device,  $R_t$  can be set just one-delayed error term  $u_{t-1}$ ,  $u_t = z^{-1}[y_t - O_t]$ , so as to avoid the difficulty to determine the lag length of recurrent input. Very small number of lag  $p$  in  $x_t$  is also appropriate for this recurrent SVR, for instance,  $p = 2$  in our application. Thus, the specification of recurrent SVR based nonlinear ARMA model used in this study is just simple ARMA (2, 1) model. It is the richer dynamic structure and specification convenience that make the recurrent SVR attractive in dynamic applications.

Now, according to equations (3)-(6), we can rewrite the primal problem of recurrent  $\varepsilon$ -SVR for nonlinear ARMA (2,1) model as follows:

$$\min_{w, b, \xi^{(t)}} C(w, b, \xi^{(t)}) = \frac{1}{2} \|w\|^2 + C \sum_{t=1}^T (\xi_t + \xi'_t) \quad (20)$$

$$s.t. \quad w^\top \phi(y_{t-1}, y_{t-2}, u_{t-1}) + b - y_t \leq \varepsilon + \xi_t, \quad (21)$$

$$y_t - w^\top \phi(y_{t-1}, y_{t-2}, u_{t-1}) - b \leq \varepsilon + \xi'_t, \quad (22)$$

$$\xi_t \geq 0, \xi'_t \geq 0, \quad t = 1, 2, \dots, T. \quad (23)$$

Also, the convex quadratic programming and linear restrictions ensure that the recurrent  $\varepsilon$ -SVR can always obtain the global unique optimal solution  $w^*$ . By using the Karush-Kuhn-Tucker conditions, we can construct its dual problem, obtain the corresponding solution,  $\alpha_t$  and  $\alpha'_t$ , and compute  $w^*$  and  $b^*$ . Because the inner-product kernel is Gaussian kernel, the regression decision function  $f(x)$  of recurrent  $\varepsilon$ -SVR is formulated as

$$\begin{aligned}
f(x) &= f(y_{s-1}, y_{s-2}, u_{s-1}) = w^{*T} \phi(y_{s-1}, y_{s-2}, u_{s-1}) + b^* \\
&= \sum_{t=1}^T (\alpha'_t - \alpha_t) \exp\left(-\frac{1}{2\sigma^2} \|(y_{s-1}, y_{s-2}, u_{s-1}) - (y_{t-1}, y_{t-2}, u_{t-1})\|^2 + b^*\right) \quad (24)
\end{aligned}$$

where  $s$  is any time point within or out of the training period. And the MA part,  $u_{s-1}$ , can be skipped and only the AR part is used for forecasting during the test period. The real constant coefficient  $\sigma^2$  is also chosen by

the users. Using the estimated decision function 24, we can achieve the best generalization capability in forecasting  $y$  on new inputs.

The difficulty to estimate the recurrent  $\varepsilon$ -SVR is that the error term is unobservable. To overcome it, we employ the model residuals as estimates of the errors in an iterative way, which is similar to the way that linear ARMA model is iteratively estimated by MLE ((Box, Jenkins & Reinsel, 1994),(Hamilton, 1997)). Likewise, we initially set the error term to be its expectation, 0. In the following, the empirical procedure of the recurrent  $\varepsilon$ -SVR executed during the training phase is described. As denoted above, the empirical procedure is illustrated for the case of the nonlinear stochastic ARMA (2, 1) model,  $y_t = g(y_{t-1}, y_{t-2}, e_{t-1}) + e_t$ . The letter  $i$  indicates the iterative epoch and  $t$  denotes the period.

*Step1: Set  $i = 1$  and star with all residuals at zero:  $e_t^{(1)} = 0$ .*

*Step2: Run a SVR procedure to get the decision function  $f^{(i)}$  to the points  $\{x_t, y_t\}$  with all inputs  $x_t = \{y_{t-1}, y_{t-2}, e_{t-1}^{(i)}\}$ .*

*Step3: Compute the new residuals  $e_t^{(i+1)} = y_t - f^{(i)}$ .*

*Step4: Terminate the computational process when the stopping criterion is satisfied; otherwise, set  $i = i + 1$  and go back to Step 2.*

Note that the first iterative epoch is in fact a feedforward SVR process and results in a AR (2) model and that the following epochs provide results of the ARMA (2,1) model, being estimated by the recurrent  $\varepsilon$ -SVR.

In general, the procedure cannot be shown to converge, and there are no well-defined criteria for stopping its operation. Rather, some reasonable criteria can be found, although with its own practical drawback, which may be used to terminate the computational process.

To formulate such a criterion, it is logical to think in terms of the properties of the estimated residual series. After the enough long iterative steps, the autocorrelation displayed behind the residuals during the first AR epoch should disappear, and the information in the residual behavior has been used out and the final residual series should be white noisy. Accordingly, we may suggest a sensible convergence criterion for the recurrent  $\varepsilon$ -SVR procedure as follows:

*The recurrent  $\varepsilon$ -SVR procedure is considered to have converged when the corresponding residuals become white noisy, or has no autocorrelation.*

To quantify the measurement of white noise, we use the formal hypothesis test, Ljung-Box-Pierce Q-test to investigate a departure from randomness based on the ACF of the residuals. Under the null hypothesis of no autocorrelation in residuals, the Q-test statistic is asymptotically Chi-Square distributed. Concretely, we just check the actual  $p$  values of Q-test of lag 1. It's reasonable to think there is no higher order autocorrelation if no one-order autocorrelation in residuals. Only if the  $p$  values of Q-test for consecutive five epochs are simultaneously higher than 0.1, the iterative computational process is stopped. To overcome the drawback of this convergence criterion, we use cross validation to avoid the possible over-fitting problem; see Subsection 4.1 for detailed information.

### 3 Empirical modeling and forecasting scheme

#### 3.1 Empirical models and their specification

As denoted in Section 2.2, very few lag numbers of nonlinear ARMA model are enough for recurrent  $\varepsilon$ -SVR and ANN approaches to capture the dynamic characteristics of data sets. Therefore, the basic forecasting framework in this study is the ARMA (2, 1) model. For the convenience of comparison, we make use of the ARMA model with the same lag orders for its linear form. The linear ARMA (2, 1) model estimated by MLE is described as below:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + e_t + \theta_1 e_{t-1} \quad (25)$$

The empirical models for the recurrent  $\varepsilon$ -SVR and the recurrent ANN are specified as the nonlinear ARMA (2, 1) process, expressed below:

$$y_t = g(y_{t-1}, y_{t-2}, e_{t-1}) + e_t \quad (26)$$

Then, the feedforward  $\varepsilon$ -SVR corresponds to the nonlinear AR (2) model,

$$y_t = g(y_{t-1}, y_{t-2}) + e_t \quad (27)$$

In this paper, the nonlinear function  $g(\cdot)$  specified for recurrent  $\varepsilon$ -SVR is radial basis function because only Gaussian kernel is chosen for SVR in this studies. Of course, other functions such as polynomial, spline, hyperbolic tangent kernel also satisfy Mercer's conditions and can be adopted as the nonlinear function of SVR. Before implement of the recurrent  $\varepsilon$ -SVR, their free parameters,  $\varepsilon$ (or denoted epsilon),  $C$  and Gaussian kernel width  $\sigma^2$ (or sigma2) must be determined in advance through cross validation. The process

of sensitivity analysis will be illustrated by using simulation in Subsection 4.1.

The benchmark recurrent ANN used in this study is the feedback multilayer perceptrons network, denoted recurrent MLP. We specify this kind of recurrent back-propagation network with the following architecture: One nonlinear hidden layer with four neurons, each using a tan-sigmoid differentiable transfer function to generate the output, and one linear output layer with one neuron. Thus, the nonlinear function  $g(\cdot)$  specified for recurrent MLP is tan-sigmoid function. As a training algorithm, the fast training Levenberg-Marquardt algorithm is chosen. The value of the learning rate parameter used in the training process is set to be 0.05. These specifications and choices are standard in neural network literatures.

### 3.2 Forecasting scheme and evaluation metrics

In this paper, a recursive forecasting scheme is employed with an updating sample window; the estimating and forecasting process is carried out recursively by updating the sample with one observation each time, re-running the recurrent  $\varepsilon$ -SVR procedure and recalculating the model parameters and corresponding forecasts. The notations used in this study are as follows; The total number of series  $y_t$  is denoted as  $T$  and the number of observations used for the first in-sample estimation is  $T_1$  (or called training sample). Then,  $T - T_1$  observations are retained as a forecasting or test sample. Let the actual series at period  $t + j$  and the  $j$ -step-ahead forecast of the series made at period  $t$  be written as  $y_{t+j}$  and  $\hat{y}_{t+j}$ , respectively. Then, we can write

$$\hat{y}_{t+j|t} = E(y_{t+j} | y_t, y_{t-1}, \dots, y_1) \quad (28)$$

so that the  $j$ -step-ahead forecast of the series made at time  $t$  is the expected value of the series  $j$  periods in the future, given all information available at time  $t$ . In equation 28,  $t = T_1, \dots, T - j$ . Thus, the forecast horizon is fixed at  $j$  steps ahead, and the starting point  $t$  is varied. Therefore, we can estimate and forecast the recurrent  $\varepsilon$ -SVR based ARMA (2, 1) model for  $n = T - j - T_1 + 1$  times.

In this paper, only one-step-ahead forecasts are used for out-of-sample forecasting evaluation which indicates  $j = 1$ . We set  $n = 100$  for linear ARMA simulation and real data,  $n = 400$  for nonlinear Lorenz simulated series.

Two evaluation metrics are employed to compare the forecasting performance among the recurrent  $\varepsilon$ -SVR and the three competing methods: normalized mean square error (NMSE) and correct sign predictions (sign)

((Pesaran & Timmerman, 1990); (Moosa, 2000)).

The NMSE measures the magnitude of the forecasting error and the sign measures correctness in predicted directions, i.e., the turning point correctness. Their formulas are

$$NMSE(\%) = 100 \times \frac{MSE}{Var(y)} = 100 \times \frac{\sum_{i=1}^n (y_i - \hat{y}_t)^2 / n}{\sum_{i=1}^n (y_t - \bar{y}_t)^2 / (n-1)} \quad (29)$$

$$sign(\%) = \frac{100}{n} \sum_{i=1}^n \alpha_i, \text{ where } \alpha_t = \begin{cases} 1 & (y_{i+1} - y_i)(\hat{y}_{i+1} - \hat{y}_i) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (30)$$

## 4 Forecasting application with simulated and real data

### 4.1 Simulations

#### Data generating process

To evaluate the forecasting performance of recurrent  $\varepsilon$ -SVR approach, we first conduct the following simulation. The target variable  $y_t, t = 1, \dots, T$  is randomly generated from two models:

1. a linear ARMA (2, 1) model:

$$y_t - 0.9y_{t-1} + 0.3y_{t-2} = e_t - 0.7e_{t-1} \quad (31)$$

where the noise inputs,  $e_t$ , are generated from the standard normal distribution and the simulated  $y_t$  are discrete;

2. a nonlinear Lorenz feedback system:

$$\begin{aligned} dy/dt &= 16(x - y) \\ dx/dt &= -yz + 45.92y - x \\ dz/dt &= yx - 4z \end{aligned} \quad (32)$$

where the step size is 0.01. The Student's  $t$  noise is included in the simulated continuous  $y_t$  series (see Lorenz (1963) for more). We include both linear and nonlinear simulation to see how the recurrent  $\varepsilon$ -SVR procedure performs when a linear series is not really applicable. In the simulations, the sample size  $T$  is 1000, and the number of replications is 200. The reported results are the mean values of 200 independent replications.



### Parameters selection and iterative epochs of recurrent SVR

The use of cross-validation is appealing particularly when we have to design a somewhat complex approach with good generalization as the goal. For example, here, we may use cross-validation to determine the values of free parameters with the best performance, and when it is best to stop training, as described in the following. The first training data, that is, the former 900 observations for linear ARMA series (briefly denoted LARMA) and 600 for nonlinear Lorenz series is exemplified. The training data is further randomly partitioned into two disjoint subsets: estimating sample and validating sample (700 and 200 observations for LARMA; 500 and 100 for Lorenz).

As shown in Section 2, two free parameters ( $\varepsilon$  and  $C$ ) and kernel width  $\sigma^2$  have to be determined by us before running the recurrent  $\varepsilon$ -SVR procedure. The motivation of using cross validation here is to validate the model on a data set different from the one used for parameter estimation. In this way we may use the training set to assess the performance of various values of parameters, and thereby choose the best one. The sensitivity analysis of recurrent  $\varepsilon$ -SVR (represented by the generalization error NMSE) with respect to three parameters are illustrated in Figure 4.

Figure 4(a)-(c) describe the sensitivity analysis for one of 200 simulated linear ARMA series. Parameter  $\varepsilon$  varies between the range

$$[0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0]$$

with  $C$  being fixed to be 0.1 and  $\sigma^2$  be 1. The values of  $\varepsilon$  before the point of  $\varepsilon$  be 0.01 have no influence on the performance of our recurrent SVR, which is considerably stable. Parameter  $C$  varies from very small value 0.0001 to infinity with  $\varepsilon$  being fixed to be 0.01 and be 1. Clearly, when  $C = 0.1$ , NMSE of validation sample obtain the lowest value, 99%; after that, over-fitting the training set occurs. Coefficient  $\sigma^2$  varies between values of 0.01 and 0.1 with  $C$  being fixed to be 0.1 and be 0.01. Both values of NMSE attain the minima when  $\sigma^2 = 1.0$ . Thus, the appropriate parameters of recurrent SVR for linear ARMA series are:  $\varepsilon = 0.01$ ,  $C = 0.1$  and  $\sigma^2 = 1.0$ . Figure 4(d)-(f) describe the parameter selection process for nonlinear Lorenz series. Similar to LARMA, the performance of recurrent SVR is very stable and not influenced by any value of  $\varepsilon$  before the point  $\varepsilon = 0.3$ . And when  $C = 50$  and  $\sigma^2 = 10$ , the values of NMSE for validation subsets all reach to their minima, 0.046%. Therefore, the correct parameters chosen for Lorenz series are  $\varepsilon = 0.1$ ,  $C = 50$  and  $\sigma^2 = 10$ , respectively.

With good forecasting performance as the goal, it is very difficult to figure out when it is best to stop training only in terms of fitting performance. It is possible for the procedure to end up over-fitting the training data if the

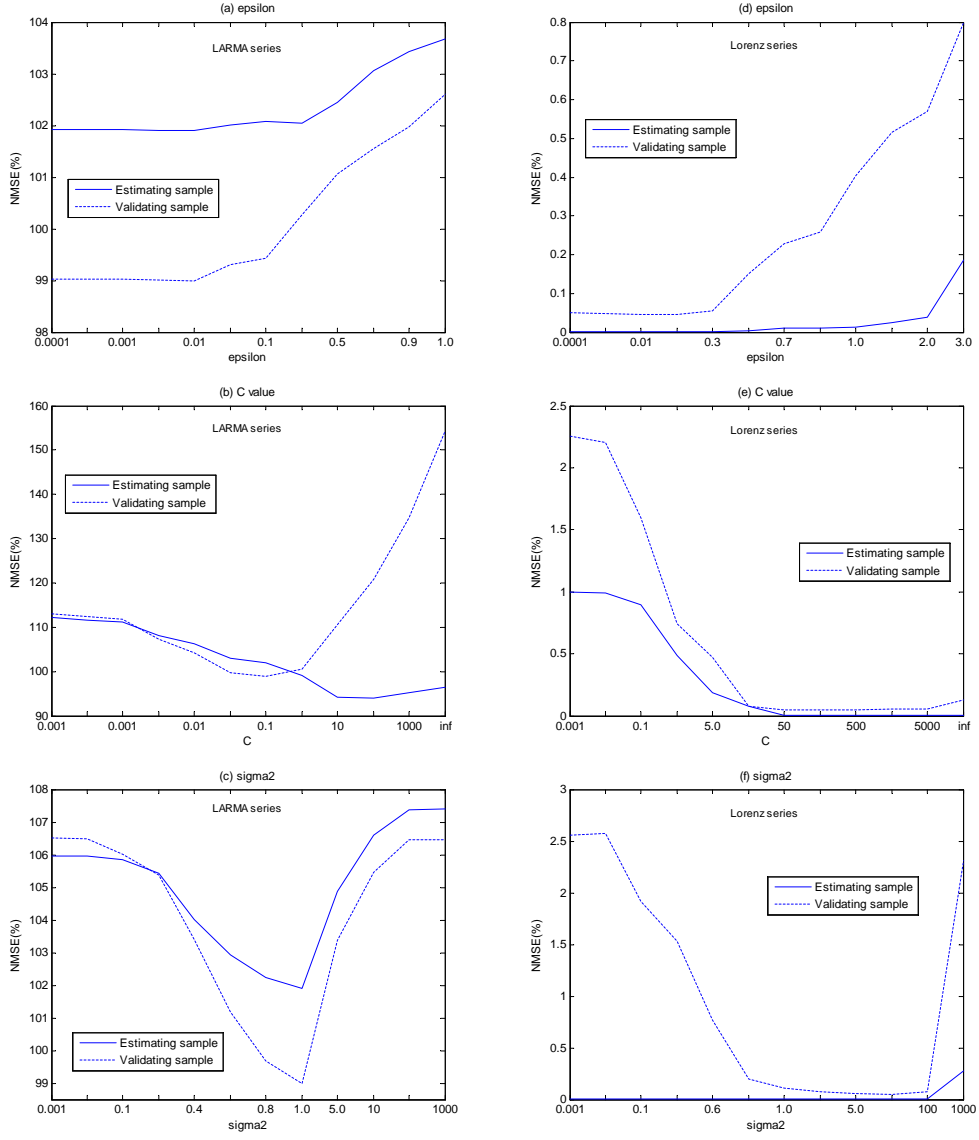


Figure 4: Sensitivity analysis of the recurrent  $\varepsilon$ -SVR for simulation data.

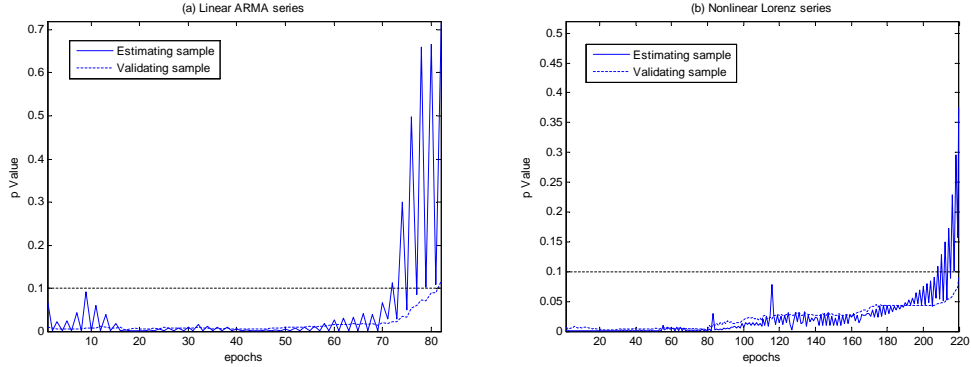


Figure 5: Iterative epochs of recurrent  $\varepsilon$ -SVR for simulation data.

training session is not stopped at the right point. We can identify the onset of over-fitting and the stopping point through the use of cross-validation. Figure 5(a) and (b) describe the iterative epochs for one of 200 linear ARMA and nonlinear Lorenz series, respectively. For the former series, the iterative process of recurrent  $\varepsilon$ -SVR is stopped at the 82th epoch; while, for the latter series, the iterative process is longer and stopped after 220 iterative steps, maybe due to the nonlinearity and noise of the series. Now, we can say, at about the 10 percent significant level, the final residuals obtained from the recurrent SVR procedure have no autocorrelation. In addition, the  $p$ -value curves of both estimating and validating samples exhibit the similar pattern (increase for an increasing number of epochs) and point to the almost same stopping point. That is to say, there is no over-fitting phenomenon for the examples illustrated here, the recurrent  $\varepsilon$ -SVR model do as well on the validating subset as it does on the estimating subset, on which its design is based.

### Comparing forecasting performance

There is still the possibility of over-fitting after training. Therefore, the generalization performance of the competed models is further measured and evaluated on the test set, which is different from the validation subset. For the simulated data, the forecasting sample is the later 100 observations for the LARMA series and later 400 for Lorenz series. Thus, the recurrent  $\varepsilon$ -SVR and the benchmark models should be recursively trained and forecasted 100 and 400 times for respective series and obtain the corresponding one-step-ahead forecasts for evaluation.

The average NMSE and the proportion of correct sign predictions of 200 replicable simulations for each method are reported in Table 1, in which, a

series	metrics	MLE	Recurrent MLP	Feedforward SVR	Recurrent SVR
LARMA	NMSE	101.23	101.05	100.97	100.96
	sign	42.91	40.37	40.16	41.76
Lorenz	NMSE	0.00624	0.00082	0.00092	0.00074
	sign	98.77	96.41	98.99	99.48

Table 1: Measures of forecasting performance for simulation data.

smaller NMSE and a higher sign value indicate the better forecasting performance. As seen in Table 1, the recurrent SVR almost outperform the benchmarks in one-step-ahead forecasting, except for sign value in LARMA series. The overall superiority of the recurrent SVR over the feedforward one reveals that the proposed recurrent  $\varepsilon$ -SVR in this study really improves the forecasting performance of the standard SVR. The fact that the recurrent SVR behaves better than the recurrent MLP but the feedforward SVR does not (evidence from the sign value for LARMA and NMSE for Lorenz) again confirms that the structural risk minimizing principle endows SVR with stronger forecasting ability as opposed to ANN and the recurrent networks has higher ability to capture the dynamic feature of series than does the feedforward one. In addition, the recurrent SVR has the similar forecasting performance to MLE for LARMA series but much better than MLE for Lorenz series (note that the values of NMSE are 0.0062 and 0.00074 for MLE and recurrent SVR, the magnitude of which is totally different). Therefore, the linear ARMA model is not suitable to the data with high nonlinearity. The far lower values of NMSE for Lorenz series than those for LARMA may be due to the continuous nature for the former and discrete the latter.

## 4.2 Real data analysis

In this subsection, we investigate the forecasting performance of all candidates by using real data for two kinds of financial variables: CAD/USD exchange rates (rates) and NYSE (NYSE) average index.

### Data description

The first data set consists of the daily nominal bilateral exchange rates of the Canadian Dollar (CAD) against the U.S. dollar for the period between January 6, 2004 and December 31, 2007. The data are obtained from a database of Policy Analysis Computing and Information Facility in Commerce (PA-

CIFIC) at the University of British Columbia. The second data set consists of daily closing price of New York Stock Exchange <sup>TM</sup>(NYSE) composite stock index for the period of January 9, 2004 to December 31, 2007. The data are downloaded directly from the Market Information section of the NYSE <sup>TM</sup>web page.

It has been widely accepted that a variety of financial variables including foreign exchange rates and stock prices are integrated of order one. To avoid the issue of possible nonstationarity, this paper considers the financial returns,  $y_t$ , which are converted from corresponding levels (price or index),  $I_t$ , by using continuous compounding transforms as

$$y_t = 100 \times (\log I_{t+1} - \log I_t) \quad (33)$$

Both data are transformed into daily returns via equation 33, providing a series of returns for 1000 observations. Same as the LARMA simulation, the recursive training is used with updating window data starting from the former 900 observations through the former 999 observations and obtain the 100 one-day-ahead forecasts of returns.

### Comparing forecasting performance

The implementation of parameter selection and iterative process of recurrent  $\varepsilon$ -SVR for real data are same as the simulations and skipped here to save space. Based on such kind of sensitivity analysis, the appropriate parameters are  $\varepsilon = 0.005$ ,  $C = 0.001$  and  $\sigma^2 = 1$  for CAD returns and  $\varepsilon = 0.3$ ,  $C = 0.01$  and  $\sigma^2 = 0.2$  for NYSE returns.

series	metrics	MLE	Recurrent MLP	Feedforward SVR	Recurrent SVR
CAD	NMSE	100.32	100.06	99.52	99.49
	sign	62.63	62.63	58.59	63.64
NYSE	NMSE	99.42	99.31	99.41	99.24
	sign	67.68	61.62	68.69	70.71

Table 2: Measures of forecasting performance for real data.

By using this specification, the recurrent  $\varepsilon$ -SVR is adopted to estimate and forecast the financial returns recursively and its forecasting performance, with the candidates, based on two quantitative metrics (NMSE and sign) are presented in Table 2. First, we compare the forecasting accuracy of the feedforward and recurrent SVR. The results are the same as simulation and

say again that the presence of the feedback loops in the standard SVR process has a positive impact on its forecasting capability. Next, we compare the forecasting performance between the recurrent SVR, MLE and the recurrent MLP. Both metrics all reveal that recurrent SVR consistently outperform the benchmarks in forecasting one-day-ahead financial returns.

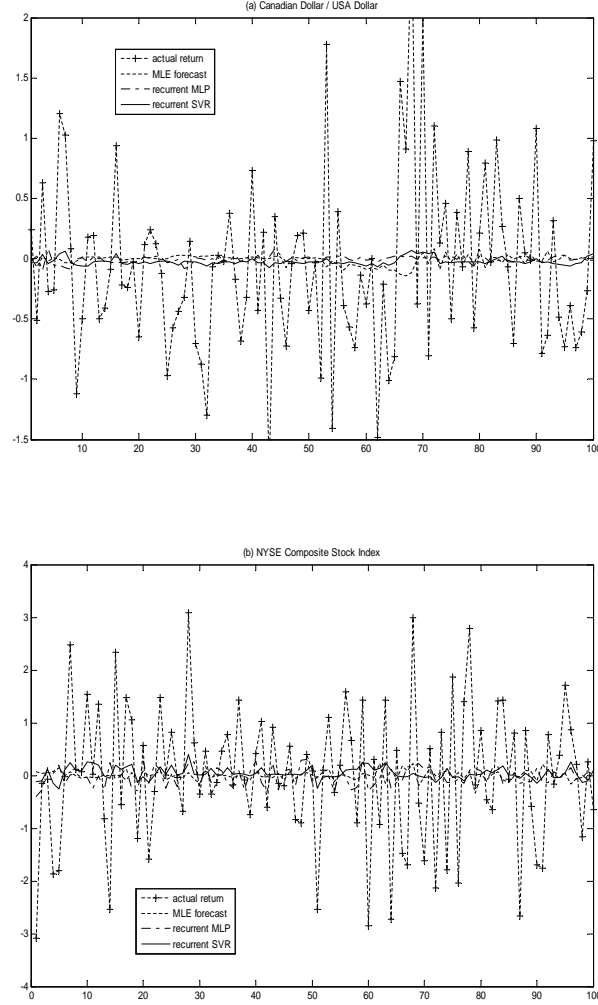


Figure 6: Actual and Forecasted Financial Returns.

We plot the actual and one-day-ahead forecasting returns by MLE, the recurrent MLP and the recurrent  $\varepsilon$ -SVR. The 100 one-day-ahead forecasting returns correspond to the out-of-sample period between August 7, 2007 and December 31, 2007 for the CAD and between August 9, 2007 and December

31, 2007 for the NYSE. As anticipated, the recurrent  $\varepsilon$ -SVR captures the actual returns more accurately.

## 5 Conclusions

In this paper we propose a recurrent  $\varepsilon$ -SVR procedure for nonlinear ARMA models which has a global feedback loop from the output layer to the input space and examine the empirical forecasting performance of the proposed procedure. Empirical applications are made for forecasting the simulated data and the real data of the Canadian Dollar (CAD) against U.S. Dollar daily exchange rates and the New York Stock Exchange <sup>TM</sup>(NYSE) composite stock index. The forecasting ability of the recurrent  $\varepsilon$ -SVR is also compared with those of MLE, the recurrent ANN (MLP) and the feedforward SVR with regard to two quantitative evaluation metrics.

The NMSE and sign evidence from both the simulated and real data analysis obviously shows that the proposed recurrent  $\varepsilon$ -SVR improves the forecasting performance of the standard feedforward one. And it also consistently outperforms the MLE and the recurrent ANN in forecasting the return magnitude and the turning points just with only a few exceptions. Empirical analysis is in favor of the theoretical advantage of the recurrent SVR. The sensitivity to free parameters of the recurrent  $\varepsilon$ -SVR results and its iterative process are also examined in detail by using cross-validation method, which can be implemented very easily. In conclusion, the proposed recurrent  $\varepsilon$ -SVR can be used as another standard SVR construction procedure in other applications.

## References

- Adya, M. and F. Collopy, 1998: How effective are neural networks at forecasting and prediction? a review and evaluation. *Journal of Forecasting.*, **17**, 481–495.
- Ashok, K. and A. Mitra, 2002: Forecasting daily foreign exchange rates using genetically optimized neural networks. *Journal of Forecasting.*, **21**, 501–511.
- Box, G., G. Jenkins, and G. Reinsel, 1994: *Time Series Analysis: Forecasting and Control*. Englewood Cliffs, Prentice Hall, New Jersey.
- Cao, L. and F. Tay, 2001: Financial forecasting using support vector machines. *Computation and Application*, **10**, 184–192.

- Deng, N. and Y. Tian, 2004: *New Methods in Data Mining: Support Vector Machine*. Science Press, Beijing.
- Espinoza, M., J. Suykens, and B. D. Moor, 2006: Ls-svm regression with autocorrelated errors. in Proc. of the 14th IFAC Symposium on System Identification (SYSID), 582-587.
- Evgeniou, T., T. Poggio, M. Pontil, and A. Verri, 2002: Regularization and statistical learning theory for data analysis. *Computational Statistics & Data Analysis*, **38**, 421-432.
- Gaudart, J., B. Giusiano, and L. Huiart, 2004: Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data. *Computational Statistics & Data Analysis*, **44**, 547-570.
- Gestel, T., J. Suykens, D. Baestaens, A. Lambrechts, G. Lanckriet, B. Vandaele, B. D. Moor, and J. Vandewalle, 2001: Financial time series prediction using least squares support vector machines within the evidence framework. *IEEE Transactions on Neural Networks*, **12**(4), 809-821. Special Issue on Neural Networks in Financial Engineering.
- Hamilton, J., 1997: *Time Series Analysis*. Princeton University Press.
- Härdle, W., R. Moro, and D. Schäfer, 2005: Predicting bankruptcy with support vector machines. *Statistical Tools for Finance and Insurance*. Springer Verlag, Berlin).
- Härdle, W., R. Moro, and D. Schäfer, 2006: Graphical data representation in bankruptcy analysis. *Handbook for Data Visualization*. Springer Verlag, Berlin).
- Haykin, S., 1999: *Neural Networks: a comprehensive foundations*. Prentice Hall, New Jersey.
- Kamruzzaman, J. and R. Sarker, 2004: Ann-based forecasting of foreign currency exchange rate. *Neural Information Processing: Letters and Reviews*, **3**(2).
- Kanas, A., 2003: Non-linear forecasts of stock returns. *Journal of Forecasting*, **22**(4), 299-315.
- Kuan, C.-M., 1995: A recurrent newton algorithm and its convergence properties. *IEEE Transactions on Neural Networks*, **6**, 779-783.



- Kuan, C.-M., K. Hornik, and H. White, 1994: A convergence result for learning in recurrent neural networks. *Neural Computation*, **6**, 420–440.
- Kuan, C.-M. and T. Liu, 1995: Forecasting exchange rates using feedforward and recurrent neural networks. *Journal of Applied Econometrics*, **10**, 347–364.
- Lee, T., C. Chiu, Y. Chou, and C. Lu, 2006: Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics & Data Analysis*, **50**(4), 1113–1130.
- Lisi, F. and R. A. Schiavo, 1999: A comparison between neural networks and chaotic models for exchange rate prediction. *Computational Statistics & Data Analysis*, **30**(1), 87–102.
- Lorenz, E., 1963: Deterministic non-periodic flow. *Journal of the Atmospheric Sciences*, **20**, 130–141.
- Moosa, I., 2000: *Exchange Rate Forecasting: Techniques and Applications*. Macmillan Press LTD, London.
- Niemira, M. and P. Klein, 1994: *Forecasting Financial and Economic Cycles*. John Wiley & Sons, New York.
- NYSE Database of nyse stock index. <http://www.nyse.com/marketinfo/datalib/>, the Market Information section of the NYSE TM web page.
- Pesaran, M. and A. Timmerman, 1990: The statistical and economic significance of the predictability of excess returns on common stocks. Department of Applied Economics, University of Cambridge, Working Paper No. 9022.
- Priestley, M., 1988: *Nonlinear and Non-stationary Time Series Analysis*. Academic Press, London, first edition.
- rates Database of exchange rates. <http://pacific.commerce.ubc.ca/xr>, Policy Analysis Computing and Information Facility In Commerce (PACIFIC) at University of British Columbia.
- Scholkopf, B. and A. Smola, 2001: *Learning with Kernels*. Cambridge, MA: MIT Press.
- Suykens, J., T. V. Gestel, J. Brabanter, B. D. Moor, and J. Vandewalle, 2002: *Least Squares Support Vector Machines*. World Scientific, Singapore.

- Suykens, J. and J. Vandewalle, 2000: Recurrent least squares support vector machines. *IEEE Transactions on Circuits and Systems-I*, **47**(7), 1109–1114.
- Tian, J., M. Juhola, and T. Grönfors, 1997: Ar parameter estimation by a feedback neural network. *Computational Statistics & Data Analysis*, **25**(1), 17–24.
- Trafalis, T. and H. Ince, 2000: Support vector machine for regression and applications to financial forecasting. International Joint Conference on Neural Networks, 348-353.
- Vapnik, V., 1995: *The Nature of Statistical Learning Theory*. Springer, New York.
- Vapnik, V., 1997: *Statistical Learning Theory*. Wiley, New York.
- Wu, B., 1995: Model-free forecasting for nonlinear time series with application to exchange rates. *Computational Statistics & Data Analysis*, **19**(4), 433–459.
- Yang, H., L. Chan, and I. King, 2002: Support vector machine regression for volatile stock market prediction. Proceedings of the Third International Conference on Intelligent Data Engineering and Automated Learning, 391-396.

## SFB 649 Discussion Paper Series 2008

For a complete list of Discussion Papers published by the SFB 649, please visit <http://sfb649.wiwi.hu-berlin.de>.

- 001 "Testing Monotonicity of Pricing Kernels" by Yuri Golubev, Wolfgang Härdle and Roman Timonfeev, January 2008.
- 002 "Adaptive pointwise estimation in time-inhomogeneous time-series models" by Pavel Cizek, Wolfgang Härdle and Vladimir Spokoiny, January 2008.
- 003 "The Bayesian Additive Classification Tree Applied to Credit Risk Modelling" by Junni L. Zhang and Wolfgang Härdle, January 2008.
- 004 "Independent Component Analysis Via Copula Techniques" by Ray-Bing Chen, Meihui Guo, Wolfgang Härdle and Shih-Feng Huang, January 2008.
- 005 "The Default Risk of Firms Examined with Smooth Support Vector Machines" by Wolfgang Härdle, Yuh-Jye Lee, Dorothea Schäfer and Yi-Ren Yeh, January 2008.
- 006 "Value-at-Risk and Expected Shortfall when there is long range dependence" by Wolfgang Härdle and Julius Mungo, January 2008.
- 007 "A Consistent Nonparametric Test for Causality in Quantile" by Kiho Jeong and Wolfgang Härdle, January 2008.
- 008 "Do Legal Standards Affect Ethical Concerns of Consumers?" by Dirk Engelmann and Dorothea Kübler, January 2008.
- 009 "Recursive Portfolio Selection with Decision Trees" by Anton Andriyashin, Wolfgang Härdle and Roman Timofeev, January 2008.
- 010 "Do Public Banks have a Competitive Advantage?" by Astrid Matthey, January 2008.
- 011 "Don't aim too high: the potential costs of high aspirations" by Astrid Matthey and Nadja Dwenger, January 2008.
- 012 "Visualizing exploratory factor analysis models" by Sigbert Klink and Cornelia Wagner, January 2008.
- 013 "House Prices and Replacement Cost: A Micro-Level Analysis" by Rainer Schulz and Axel Werwatz, January 2008.
- 014 "Support Vector Regression Based GARCH Model with Application to Forecasting Volatility of Financial Returns" by Shiyi Chen, Kiho Jeong and Wolfgang Härdle, January 2008.
- 015 "Structural Constant Conditional Correlation" by Enzo Weber, January 2008.
- 016 "Estimating Investment Equations in Imperfect Capital Markets" by Silke Hüttel, Oliver Mußhoff, Martin Odening and Nataliya Zynych, January 2008.
- 017 "Adaptive Forecasting of the EURIBOR Swap Term Structure" by Oliver Blaskowitz and Helmut Herwatz, January 2008.
- 018 "Solving, Estimating and Selecting Nonlinear Dynamic Models without the Curse of Dimensionality" by Viktor Winschel and Markus Krätzig, February 2008.
- 019 "The Accuracy of Long-term Real Estate Valuations" by Rainer Schulz, Markus Staiber, Martin Wersing and Axel Werwatz, February 2008.
- 020 "The Impact of International Outsourcing on Labour Market Dynamics in Germany" by Ronald Bachmann and Sebastian Braun, February 2008.
- 021 "Preferences for Collective versus Individualised Wage Setting" by Tito Boeri and Michael C. Burda, February 2008.

**SFB 649, Spandauer Straße 1, D-10178 Berlin**  
**<http://sfb649.wiwi.hu-berlin.de>**

This research was supported by the Deutsche  
Forschungsgemeinschaft through the SFB 649 "Economic Risk".



- 022 "Lumpy Labor Adjustment as a Propagation Mechanism of Business Cycles" by Fang Yao, February 2008.
- 023 "Family Management, Family Ownership and Downsizing: Evidence from S&P 500 Firms" by Jörn Hendrich Block, February 2008.
- 024 "Skill Specific Unemployment with Imperfect Substitution of Skills" by Runli Xie, March 2008.
- 025 "Price Adjustment to News with Uncertain Precision" by Nikolaus Hautsch, Dieter Hess and Christoph Müller, March 2008.
- 026 "Information and Beliefs in a Repeated Normal-form Game" by Dietmar Fehr, Dorothea Kübler and David Danz, March 2008.
- 027 "The Stochastic Fluctuation of the Quantile Regression Curve" by Wolfgang Härdle and Song Song, March 2008.
- 028 "Are stewardship and valuation usefulness compatible or alternative objectives of financial accounting?" by Joachim Gassen, March 2008.
- 029 "Genetic Codes of Mergers, Post Merger Technology Evolution and Why Mergers Fail" by Alexander Cuntz, April 2008.
- 030 "Using R, LaTeX and Wiki for an Arabic e-learning platform" by Taleb Ahmad, Wolfgang Härdle, Sigbert Klinke and Shafeeqah Al Awadhi, April 2008.
- 031 "Beyond the business cycle – factors driving aggregate mortality rates" by Katja Hanewald, April 2008.
- 032 "Against All Odds? National Sentiment and Wagering on European Football" by Sebastian Braun and Michael Kvasnicka, April 2008.
- 033 "Are CEOs in Family Firms Paid Like Bureaucrats? Evidence from Bayesian and Frequentist Analyses" by Jörn Hendrich Block, April 2008.
- 034 "JBendge: An Object-Oriented System for Solving, Estimating and Selecting Nonlinear Dynamic Models" by Viktor Winschel and Markus Krätzig, April 2008.
- 035 "Stock Picking via Nonsymmetrically Pruned Binary Decision Trees" by Anton Andriyashin, May 2008.
- 036 "Expected Inflation, Expected Stock Returns, and Money Illusion: What can we learn from Survey Expectations?" by Maik Schmeling and Andreas Schrimpf, May 2008.
- 037 "The Impact of Individual Investment Behavior for Retirement Welfare: Evidence from the United States and Germany" by Thomas Post, Helmut Gründl, Joan T. Schmit and Anja Zimmer, May 2008.
- 038 "Dynamic Semiparametric Factor Models in Risk Neutral Density Estimation" by Enzo Giacomini, Wolfgang Härdle and Volker Krätschmer, May 2008.
- 039 "Can Education Save Europe From High Unemployment?" by Nicole Walter and Runli Xie, June 2008.
- 042 "Gruppenvergleiche bei hypothetischen Konstrukten – Die Prüfung der Übereinstimmung von Messmodellen mit der Strukturgleichungsmethodik" by Dirk Temme and Lutz Hildebrandt, June 2008.
- 043 "Modeling Dependencies in Finance using Copulae" by Wolfgang Härdle, Ostap Okhrin and Yarema Okhrin, June 2008.
- 044 "Numerics of Implied Binomial Trees" by Wolfgang Härdle and Alena Mysickova, June 2008.
- 045 "Measuring and Modeling Risk Using High-Frequency Data" by Wolfgang Härdle, Nikolaus Hautsch and Uta Pigorsch, June 2008.
- 046 "Links between sustainability-related innovation and sustainability management" by Marcus Wagner, June 2008.

**SFB 649, Spandauer Straße 1, D-10178 Berlin**  
<http://sfb649.wiwi.hu-berlin.de>

This research was supported by the Deutsche  
Forschungsgemeinschaft through the SFB 649 "Economic Risk".



- 047 "Modelling High-Frequency Volatility and Liquidity Using Multiplicative Error Models" by Nikolaus Hautsch and Vahidin Jeleskovic, July 2008.
- 048 "Macro Wine in Financial Skins: The Oil-FX Interdependence" by Enzo Weber, July 2008.
- 049 "Simultaneous Stochastic Volatility Transmission Across American Equity Markets" by Enzo Weber, July 2008.
- 050 "A semiparametric factor model for electricity forward curve dynamics" by Szymon Borak and Rafał Weron, July 2008.
- 051 "Recurrent Support Vector Regreson for a Nonlinear ARMA Model with Applications to Forecasting Financial Returns" by Shiyi Chen, Kiho Jeong and Wolfgang K. Härdle, July 2008.

**SFB 649, Spandauer Straße 1, D-10178 Berlin**  
**<http://sfb649.wiwi.hu-berlin.de>**

This research was supported by the Deutsche  
Forschungsgemeinschaft through the SFB 649 "Economic Risk".

